Learning from High Dimensional fMRI Data using Random Projections

Author: Madhu Advani

December 16, 2011

Introduction

The term "the Curse of Dimensionality" refers to the difficulty of organizing and applying machine learning to data in a very high dimensional space. The reason for this difficulty is that as the dimensionality increases, the volume between different training examples increases rapidly and the data becomes sparse and difficult to classify. So, the predictive power of a machine learning algorithm decreases as the dimensionality increases with a fixed number of training examples, which is known as the Hughes effect.

One way of dealing with the curse of dimensionality is by projecting data into a lower-dimensional subspace. The statistically optimal way to do this (assuming the data is on or near a linear subspace) is PCA, which projects the data into a subspace that preserves as much of the variation as possible. However, PCA is computationally expensive for high-dimensional data compared to the method of dimensionality reduction through random projections.

In fact, the distortion of the data when it is compressed via random projections can be bound by the JL lemma. However, empirical testing of this technique will demonstrate how well it performs in practical machine learning problems. The major benefit of using random projections are that they are a computationally less expensive than PCA particulary when the dimensionality of the data becomes too large for matrix diagonalization to be feasible, as is the case for very high-dimesionality data such as fMRI.

The main focus of this paper is applying machine learning algorithms to classify fMRI data and attempting to empirically and theoretically predict the feasibility of applying random projections to the supervised classifying of fMRI data. To that end, we first give a review of some of the applicable theory for random projections, then we empirically examine how logistic regression classification error varies when the data is compressed via random projections. We then explain a theoretical method for estimating an assymptotic bound on the generalization error, estimated using k-fold cross validation.

1 Theory - Random Projections and Machine Learning

1.1 Random Projections Preserve Relationship Between Data - JL Lemma

Random compression of data points $x_1, x_2, ..., x_P$ from N to M dimensions via random projections is computed simply by letting

$$x_i' = Ax_i \tag{1}$$

where A is a matrix $A : R^N \to R^M$ with each A_{ij} a normally distributed random variable with mean 0 and variance $1/M$. In fact the distribution need not be normal, and the JL Lemma will still hold true if we instead let each A_{ij} equal $\frac{1}{\sqrt{M}}$ or $-\frac{1}{\sqrt{M}}$ with equal probability. For computation simplicity, our results used the latter method for compression.

The JL lemma state that $M = O\left(\frac{1}{\delta^2}\log(S)\right)$ is sufficient so that with a high probability (error exponentially decaying with M) we have a bound on the distortion of two points x_i, x_j

$$\left|\frac{\|A(x_i - x_i)\|^2 - \|x_i - x_j\|^2}{\|x_i - x_j\|^2}\right| \leq \delta \tag{2}$$

Where S is the number of pairs of point (or vectors to be preserved). Here $S = O\left(P^2\right)$, so $M = O\left(\frac{1}{\delta^2}\log(P)\right)$.

A detailed proof can be found in [3], but to provide a rough sketch, we consider taking the vector between each pair of points v_i and compressing them. It is not hard to show that the mean value of the squared length of each vector has the same expected value in the compressed space, and we then use the fact that each dimension is independent to show that the we can compute a Chernoff-style bound on the distance of the compressed vector from its mean length.

1.2 Predicting Machine Learning ability using Random Projections

Now, we want to use the JL lemma to predict the effect of randomly compressing data on Machine Learning. We will assume the case of a classification problem where each data point x_i is labeled by y_i as -1 or 1. We can assume that a machine learning algorithm will separate our data equally well if the data is rotated or shifted, without altering relative distanes between points. Thus we will consider shifting and rotating our data so that the means of the data μ_{-1} and μ_1 lie on the z_1 axis, equidistant from the origin.

Now, as in [2] we will define that data is separated by a margin γ if there exists a unit length vector w that bounds the angle separating the two distributions of the data:

$$P[y\,(w \cdot x)\,/\|x\| < \gamma] = 0 \tag{3}$$

Note: we will also call $y_i\,(w \cdot x_i)\,/\|x_i\|$ the margin of a data point x_i

First consider S data points with some margin γ and randomly project this data from N to M dimensions.

For some $M = O\left(\frac{log(S)}{\delta^2}\right)$ and letting u and v be normalized vectors in our dataset

$$|u \cdot v - Au \cdot Av| \leq \frac{1}{2}\left|\|A(u-v)\|^2 - \|u-v\|^2\right| + \frac{1}{2}\left|\|Au\|^2 - 1\right| + \frac{1}{2}\left|\|Av\|^2 - 1\right| \tag{4}$$

Thus, we have a high probability that

$$|u \cdot v - Au \cdot Av| \leq \frac{\delta}{2}\left(\|u-v\|^2 + 2\right) \leq 3\delta \tag{5}$$

It is not hard to extend this result (as in [4]) to show that for sufficiently small δ

$$\left|u \cdot v - \frac{Au \cdot Av}{\|Au\|\|Av\|}\right| \leq 6\delta \tag{6}$$

Thus, if we choose M large enough such that the maximum distortion likely to be less than is $\gamma/12$, then we have

$$\left| \frac{Ax_i \cdot Aw}{\|Ax_i\|} - \frac{x_i \cdot w}{\|x_i\|} \right| \leq \gamma/2 \tag{7}$$

Thus, we can ensure the margin is reduced my no more than a factor of $\frac{\gamma}{2}$ with high probability for some $M = O\left(\frac{\log(P)}{\gamma^2}\right)$

2 Empirical Testing of Learning on High Dimensional fMRI Data

2.1 High Dimensional fMRI Data

The StarPlus fMRI data used in this paper was originally collected by Marcel Just and his colleagues in Carnegie Mellon University's CCBI. In the experiment, fMRI imaging was performed on a fraction of the brain, while the subjects were told to either stare at a blank wall or a sentence or picture, which they viewed seperately.

fMRI scans measure the level of oxygen in the the blood flow of the brian, and is correlated with brain activity in different regions of the brain. The brian of the subject was scanned every .5 seconds over a period of about half a minute for each trial and about 4 seconds of looking at a picture and 4 seconds of looking at a sentence. The results of the brain scans in each .5 second time step is stored in a vector of activity measurements about 5000 voxels (volume elements) of the brain.

The most obvious application of machine learning was to use a machine learning algorithm to classify whether the subject was looking at a picture or a sentence based on the brain activity during the course of their fMRI.

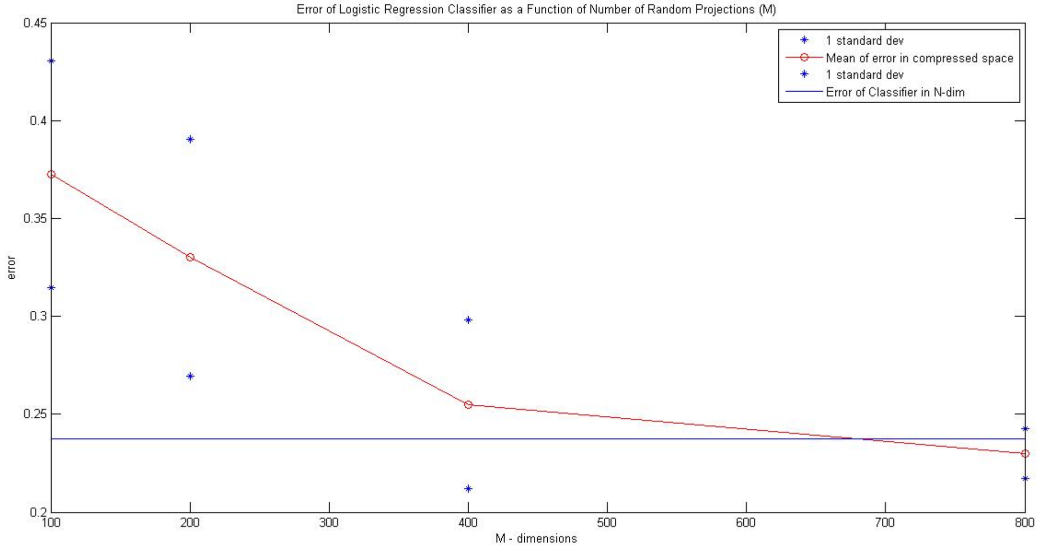
To do this I concatenated the 16 vectors of voxels collected while a subject stared at a sentence or picture for 4 seconds followed by a blank screen for 4 seconds. Then, I reduced the dimensionality of the data by restricting the data to certain regions of interest suggested to be indicative. Thus learning was performed on 80 training examples, each 27,440 dimensional with an equal number of examples corresponding to sentence and picture viewing.

2.2 Machine Learning and Results

Logistic regression gave an acceptable level of classification accuracy (generally between 70 and 80 percent) when applied to the full dimensional data set, and performed better than Gaussian Naive Bayes. Due to the high dimensionality of the data and low number of data points, machine learning algorithms intrinsically have a high variance, so I applied k-fold cross validation to the 80 training examples to approximate the generalization error of the algorithm.

When I randomly compressed the data and applied logistic regression I found that the mean value of the error and variability of the error decayed as I varied the number of projected dimensions M between 100 and 800, and the expected error for 800 dimension was actually lower than the expected generalization error for the full dimensional data set. Eventually, we would expect the error to rise for some larger M as the JL lemma predicts that the two results will converge when M becomes large enough.

Figure 1: Plot of Cross Validation Error as a Function of Number of Random Dimensions M



3 Discussion of Theortically Approximating Machine Learning Generalization Error

Using the previous theoretical ideas we attempt to approximate an assymptotic bound on the generalization error of learning from randomly compressed data. Because the data is so high-dimensional, if we apply logistic regression to the data, it results in a positive margin and zero error, which is an artifact of overfitting. So, we will estimate the generalization error using hold-one-out cross validation.

Assuming that we are running a learning algorithm in the full N-dimensional space and applying hold-one-out cross validation to test it, we will generate a classifier $\vec{w_i}$ (learned on the other $P-1$ points) for each $\vec{x_i}$. Now, by appling the results of the JL lemma for margins to the set of pairs of vectors $(\vec{w_i}, \vec{x})$ we see that we can take the fraction α of w_i and x_i that result in a margin larger than γ, and then projecting these pairs of vectors down to M dimensions, we can maintain a margin separating the data with high probability with $M = O\left(\frac{\log(P)}{\gamma^2}\right)$. If we assume that Aw, where w is the classifier learned in the high dimensional space, has a generalization error that greater than or on the order of the error of our learning algorithm on the compressed data, then we have approximately bound the error from above by $1-\alpha$ with a high probability

One indication for why margins, as defined in this paper, may be important for predicting the number of projections required is as follows. How many projections are required for a machine learning algorithm to learn from data that is compressed via random compressions is a function of the probability distribution of the data. One interesting result is that if we call let the ideal generalization error be the classifier that results in the minimum expected number of misclassifications of data and we let the data be distributed as two gaussians with an identical, isotropic variance, then we find that projection onto a single random dimension will compress both the separation between the distributions and the spread so that the generalization error remains the same. However, if we choose our variance to be non-isotropic, it is more difficult to separate in projections that lie approximately parallel to the large variance. Thus, a larger number of random projections may be required to achieve the same generalization error. This illustrates that the non-isotropic nature of the variance of data contributes to needing a larger M to accurately separate the data. This relats to our margin discussion since non-isotropic variance (in directions orthogonal to the vector between the means of the distributions) correspond to a larger fraction of data points having a low margin.

Conclusions

In Conclusion, Random projections are a useful way of reducing the dimensionality of data sets that are too high-dimensional to apply PCA to. The JL lemma guarantees preservation of stucture with large enough M, so that the generalized probability distribution will be preserved up to any degree required by large enough M. However, the number of generalized dimension it will require is really a property of the data set, one useful measure of which is the distributions of margins of the data points.

One important property of high-dimensional data is that it appears to be more seperable than it actually is. i.e. we can find a hyperplane separating a small set of data with zero error, but this not mean that the generalization error of the hyperplane is low. So predicting the effects of random projections on the generalization error of the distribution remains an interesting problem, which may require some extension of the JL lemma to accurately bound, as the JL Lemma applies to a finite set of points rather than a distribution.

Acknowledgements

Thanks to Professor Surya Ganguli for mentoring on this project

References

1) Bingham, E. Mannil, H. Random Projections in dimensionality reduction: Applications to image and text data. Helisinki University of Technology

2) Blum, A. Random Projections, Margins, Kernels, and Feature Selection. Department of Computer Science, CMU.

3) Dasgupta, S., Gupta, A. An Elementary Proof of the Johnson-Lindenstrauss Lemma. International Computer Science Institute.

4) Karnin, Z., et al. Explicit Dimension Reduction and Its Applications. Proceedings of the 26th Annual CCC. 2011